

## **REMARKS**

Claims 1-48, 50, 52, 70-73, 75-87, 89, and 91-108 were presented for examination. The Examiner objected to the oath or declaration as failing to comply with 37 CFR 1.67(a). The Examiner objected to claims 14-15, 21, 26, 28, 95, 99, 104, and 108 for failing to spell out acronyms. The Examiner objected to claims 7, 35, 45, 46, 50, and 52 for containing typographical errors. The Examiner rejected claims 20 and 24 under 35 U.S.C. §112 as indefinite for providing insufficient antecedent basis. The Examiner rejected claims 1-48, 50, 52, 70-73, 75-76, 79-87, 89, and 91-108 under 35 U.S.C. §103(a) as unpatentable over Lewallen (United States Patent No. 6,941,520, hereafter “Lewallen”) in view of Gergic et al. (United States Publication 2003/0046316, hereafter “Gergic”). The Examiner rejected claims 77-78 under 35 U.S.C. §103(a) as unpatentable over Lewallen in view of Gergic, and further in view of Snyder (United States Patent No. 6,707,475, hereafter “Snyder”).

Claims 1, 4-15, 17-24, 26-30, 35-46, 50, 52, 72-73, 75, 81-84, 87, 89, 91, 93-96, 98-100, 102, 104, and 108 have been amended. Claim 2 has been canceled. No claims were added. Claims 1, 21, 38, and 91 are independent.

Applicants’ attorneys thank the Examiner for the opportunity to discuss this application and his objections to the declarations by telephone on October 6, 2008.

### **Objection to the Oath or Declaration**

The Examiner stated that the declaration did not identify the application by application number and filing date as required by 37 CFR 1.67(a). The Examiner further stated that the declaration did not identify the mailing address of each inventor and was not executed in accordance with either 37 CFR 1.66 or 1.68. However, as confirmed by the Examiner on October 6, 2008, the supplemental Application Data Sheet filed on December 21, 2007, provides the mailing address of each inventor in satisfaction of the requirements of 37 CFR 1.63 and 1.76, and the copy of the executed declarations as filed on February 2, 2004, satisfies the requirements of 37 CFR 1.67 and of 1.68.

#### Objections to Claims 7, 14-15, 21, 26, 28, 35, 45, 46, 50, 52, 95, 99, 104, and 108

The Examiner objected to claims 14-15, 21, 26, 28, 95, 99, 104, and 108 for failing to spell out acronyms. The Examiner objected to claims 7, 35, 45, 46, 50, and 52 for containing typographical errors. Applicants hereby amend claims 7, 14-15, 21, 26, 28, 35, 45, 46, 50, 52, 95, 99, 104, and 108 to overcome these objections.

#### Rejection of Claims 20 and 24 Under 35 U.S.C. §112

The Examiner rejected claims 20 and 24 under 35 U.S.C. §112 as indefinite for providing insufficient antecedent basis. Applicants hereby amend claims 20 and 24 to overcome this rejection.

#### Rejection of Claims 1-48, 50, 52, 70-73, 75-76, 79-87, 89, and 91-108 Under 35 U.S.C. §103(a)

The Examiner rejected claims 1-48, 50, 52, 70-73, 75-76, 79-87, 89, and 91-108 under 35 U.S.C. §103(a) as unpatentable over Lewallen in view of Gergic. To establish a prima facie case of obviousness, the prior art reference (or references when combined) must teach or suggest each and every claim limitation.

The claimed invention provides a unifying data superstructure that defines appearance and behavior of an application and that contains the application state, program code and internal logic of an application, incorporating all of the aspects of a program that would normally be controlled through private data structures, procedural code, and application programming interfaces. *See* Specification, 21, and claims 1, 21, 38 and 91. Independent claims 1, 21, 38 and 91 as proposed to be amended recite, in relevant part:

updating, in accordance with the superstructure segment update,  
the application state stored in the platform-independent data  
superstructure in the device.

Lewallen fails to teach or suggest updating an application state in a platform-independent data superstructure that defines appearance and behavior **and** that stores the application state, program code and internal logic of an application, as recited by the pending claims. The focus of

Lewallen is on providing Java-based user interfaces with improved look-and-feel by extending Java programs to allow access to other objects and programs available in commonly-used user interface programs. *See* Lewallen, col. 6, lines 45-55, and col. 5, lines 30-40.

The Examiner admits that Lewallen does not disclose updating the application state in the device and relies on Gergic to cure the deficiencies of Lewallen. However, Gergic also fails to teach updating an application state stored in a platform-independent data superstructure. Gergic focuses on solving the problems faced by programmers attempting to create multi-channel applications – programs which are accessible via HTML browsers, speech-based browsers, and wireless browsers. *See* Gergic, p. 1, paragraph [0006]. Gergic provides a programming language to solve these problems, referred to as a “conversational markup language”, which allows authors to create multi-channel applications using interaction-based elements. *See* Gergic, Abstract and p. 2, paragraph [0016].

However, the programming language provided by Gergic requires separating application programs into three different aspects – content aspects, presentation aspects, and interaction aspects – so that the content of the application may be written independently of the device-specific or channel-specific presentation and interaction aspects of the program. *See* Gergic, p. 2, paragraph [0019]. Gergic states that a page written in the disclosed programming language and including the gesture-based components of interaction is connected to a separate backend including application logic, which is programmed or developed conventionally. *See* Gergic, p. 8, paragraph [0108]. Gergic fails to disclose combining the aspects of content, presentation and interaction in single superstructure and, therefore, fails to teach a platform-independent data superstructure storing an application state, program code and internal logic of an application as required by the pending claims.

Nor does Gergic suggest a platform-independent data superstructure storing an application state, program code and internal logic of an application. One of ordinary skill in the art would not use or modify Gergic to develop a system requiring the use of a superstructure encapsulating both appearance and functionality, and including the application state, program code, and internal logic of an application, because Gergic explicitly states that the goal of providing modality-independent services naturally leads to the design of programming languages

that separate information content, information presentation, and interaction logic into distinct components and that the design of languages that separate interaction logic from information content is the next evolutionary step in the design of modality-independent services. *See* Gergic, p. 4, paragraph [0031]. Gergic states that it is this separation that leads to the conversational computing approach, which provides an aggregation of modality-independent conversational gestures and which focuses on interaction alone. *See* Gergic, p. 4, paragraph [0031]. Modifying Gergic to require a superstructure encapsulating both appearance and functionality and including the application state, program code, and internal logic of an application would require changing the principle on which Gergic operates.

In contrast to Gergic, the claimed invention solves a different and broader problem than merely the presentation of data in multi-modal or multi-channel environments: enabling the creation and management of a superstructure containing a platform-independent application whose appearance and functionality is consistently propagated across heterogeneous device types for cross-device interoperability, replicability, and compatibility of applications and data with a consistent user experience, including, for example, providing solutions to problems such as distribution execution, incremental updates and process migration solutions. By focusing on only the problem of presentation, Gergic fails to teach or suggest the limitations of the pending claims, or even to teach or suggest any environment from which the claimed invention could be developed.

Therefore, neither Lewallen nor Gergic teach or suggest updating application state stored in a platform-independent data superstructure storing an application state, program code and internal logic of an application. Even if one of ordinary skill in the art focused on improving a user interface for an existing Java application modified the Lewallen system – which transforms APIs of one type into APIs of another within an existing program – to incorporate the approach in Gergic of using a programming language to generate new programs based on the expression of interactions between users and different types of devices, the combination would fail to teach or suggest the use of a superstructure storing application state, program code, and internal logic of an application.

Accordingly, Applicants submit that neither Lewallen nor Gergic, alone or in combination, disclose, teach or suggest each and every limitation of claims 1-48, 50, 52, 70-73, 75-76, 79-87, 89, and 91-108. Applicants respectfully request that the Examiner reconsider and withdraw the rejection of claims 1-48, 50, 52, 70-73, 75-76, 79-87, 89, and 91-108.

Rejection of Claims 77-78 Under 35 U.S.C. §103(a)

The Examiner rejected claims 77-78 under 35 U.S.C. §103(a) as unpatentable over Lewallen, in view of Gergic, and further in view of Snyder. To establish a prima facie case of obviousness, the prior art reference (or references when combined) must teach or suggest each and every claim limitation.

The arguments made above with respect to the failure of Lewallen and Gergic to teach or suggest the each and every limitation of the pending claims, apply with equal force here and are reiterated as if set forth in full. Snyder also fails to teach a superstructure storing application state, program code, or internal logic of an application. Snyder describes a system for selecting and displaying navigational information to provide more efficient and intuitive methods for planning and programming navigational paths. *See* Snyder, Abstract. This system for displaying navigational information does not teach a superstructure storing application state, program code, and internal logic of an application.

Nor does Snyder suggest a superstructure storing application state, program code, or internal logic of an application. Snyder provides displays for flight management systems providing pilots with more efficient, easier to use systems for displaying FMS data and procedures that relate to the arrival and departures of aircrafts. *See* Snyder, col. 2, lines 55-60. One of ordinary skill in the art would not be motivated to modify a system for intuitive display of navigational information to generate a platform-independent data superstructure defining both the appearance and behavior of an application because new methods for the creation of applications would not improve the display of navigational information for existing methods of planning navigational paths.

Accordingly, Applicants submit that neither Lewallen nor Gergic nor Snyder, alone or in combination, disclose, teach or suggest each and every limitation of claims 77-78. Applicants respectfully request that the Examiner reconsider and withdraw the rejection of claims 77-78.

## **CONCLUSION**

In view of the above, each of the presently pending claims in this application is believed to be in immediate condition for allowance. Accordingly, the Examiner is respectfully requested to pass this application to issue.

Please charge any fees associated with this filing, or apply any credits, to our Deposit Account No. 03-1721.

Respectfully submitted,  
CHOATE, HALL & STEWART LLP

Date: October 8, 2008

/John D. Lanza/  
John D. Lanza  
Registration No. 40,060

Patent Group  
CHOATE, HALL & STEWART LLP  
Two International Place  
Boston, MA 02110  
Tel: (617) 248-5000  
Fax: (617) 502-5002